

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A ~~computer-implemented~~ method for processing a p-code file, comprising:

analyzing p-code methods within said p-code file by a computer to determine a resource utilization for the p-code methods;

identifying one or more p-code methods that have a resource utilization parameter above a threshold level; ~~and~~

annotating said identified p-code methods to be compiled, said annotating comprising inserting an in-line priority level hint for each annotated p-code method, said priority level hints being hierarchically-related and collectively representing a hierarchical order, said priority level hints enabling preferential processing of said p-code methods in a hierarchical manner corresponding to said hierarchical order of said priority level hints;

replacing one or more lines of instructions in the p-code file with compiled code for the identified p-code methods; and

communicating the p-code file via a network to a target environment for execution of the compiled code and interpretation of uncompiled instructions in the p-code file.

2. (Canceled)

3. (Previously Presented) The method of claim 1, wherein:

said p-code file comprises an application for processing by a virtual machine (VM) just-in-time (JIT) compiler.

4. (Canceled)

5. (Previously Presented) The method of claim 1, further comprising providing said priority level hints as a separate file.

6. (Previously Presented) The method of claim 1, wherein:
said resource utilization parameter comprises at least one of a method execution time, a frequency of method invocation, a number of instructions and a use of loop structures.

7. (Previously Presented) The method of claim 1, wherein:
said resource utilization parameter comprises at least one of an execution time parameter, an input/output utilization parameter and a processor utilization parameter.

8-9. (Canceled)

10. (Previously Presented) The method of claim 1, wherein:
said annotating comprises selectively setting each of a plurality of normally unused bits within a method access flag field of an identified class file, wherein said unused bits are selectively set to define thereby said priority level hint of a respective annotated method.

11. (Previously Presented) The method of claim 1, wherein:

each identified p-code method is associated with one of a plurality of priority levels, said priority level hints being indicative of respective priority levels.

12. (Previously Presented) The method of claim 3, further comprising: selectively pre-compiling portions of said p-code file that are designated by the in-line hints.

13. (Previously Presented) The method of claim 12, wherein: said precompiled portion of said application file is included within a virtual machine.

14-33. (Canceled)

34. (Previously Presented) The method of claim 1, further comprising managing storage of methods in a cache memory according to the hierarchical order.

35. (Currently Amended) A computer-readable medium storing computer-executable instructions that, when executed, cause a computer to for performing the following a method for processing a p-code file, comprising:

analyzing p-code methods within said p-code file by the computer to determine a resource utilization for the p-code methods;

identifying one or more p-code methods that have a resource utilization parameter above a threshold level; ~~and~~

annotating said identified p-code methods to be compiled, said annotating comprising inserting an in-line priority level hint for each annotated p-code method, said priority level hints being hierarchically-related and collectively representing a hierarchical order, said priority

level hints enabling preferential processing of said p-code methods in a hierarchical manner corresponding to said hierarchical order of said priority level hints;

replacing one or more lines of instructions in the p-code file with compiled code for the identified p-code methods; and

communicating the p-code file via a network to a target environment for execution of the compiled code and interpretation of uncompiled instructions in the p-code file.

36. (Currently Amended) The computer-readable medium of claim 35, further storing computer-executable instructions that, when executed, cause the computer to provide said priority level hints as a separate file.

37. (Previously Presented) The computer-readable medium of claim 35, wherein: said resource utilization parameter comprises at least one of a method execution time, a frequency of method invocation, a number of instructions and a use of loop structures.

38. (Previously Presented) The computer-readable medium of claim 35, wherein: said resource utilization parameter comprises at least one of an execution time parameter, an input/output utilization parameter and a processor utilization parameter.

39. (Previously Presented) The computer-readable medium of claim 35, wherein: said annotating comprises selectively setting each of a plurality of normally unused bits within a method access flag field of an identified class file, wherein said unused bits are selectively set to define thereby said priority level hint of a respective annotated method.

40. (Previously Presented) The computer-readable medium of claim 35, wherein:
each identified p-code method is associated with one of a plurality of priority levels,
said priority level hints being indicative of respective priority levels.

41. (Currently Amended) The computer-readable medium of claim 35, further storing
computer-executable instructions that, when executed, cause the computer to perform for
~~performing the following:~~
selectively pre-compiling portions of said p-code file that are designated by the in-line hints.

42. (Cancelled)

43. (Currently Amended) The computer-readable medium of claim 35, further
storing computer-executable instructions that, when executed, cause the computer to perform
~~for performing the following:~~
managing storage of methods in a cache memory according to the hierarchical order.

44. (New) An apparatus comprising:

a processor; and

a memory storing computer-executable instructions that, when executed, cause the
apparatus to:

analyze p-code methods within said p-code file by the computer to determine a
resource utilization for the p-code methods;

identify one or more p-code methods that have a resource utilization parameter
above a threshold level;

annotate said identified p-code methods to be compiled, said annotating comprising inserting an in-line priority level hint for each annotated p-code method, said priority level hints being hierarchically-related and collectively representing a hierarchical order, said priority level hints enabling preferential processing of said p-code methods in a hierarchical manner corresponding to said hierarchical order of said priority level hints;

replace one or more lines of instructions in the p-code file with compiled code for the identified p-code methods; and

communicate the p-code file via a network to a target environment for execution of the compiled code and interpretation of uncompiled instructions in the p-code file.

45. (New) The apparatus of claim 44, wherein said annotating comprises selectively setting each of a plurality of normally unused bits within a method access flag field of an identified class file, wherein said unused bits are selectively set to define thereby said priority level hint of a respective annotated method.

46. (New) The apparatus of claim 44, wherein each identified p-code method is associated with one of a plurality of priority levels, said priority level hints being indicative of respective priority levels.

47. (New) The apparatus of claim 44, wherein the computer-executable instructions cause the apparatus to selectively pre-compile portions of said p-code file that are designated by the in-line hints.

48. (New) The apparatus of claim 44, wherein the computer-executable instructions cause the apparatus to manage storage of the p-code methods in a cache memory according to the hierarchical order.